

# WSDL RDF Mapping

Jacek Kopecký

2005/12/14



# Introduction

- WSDL 2.0 RDF Mapping
- Representation of WSDL 2.0 in RDF
  - In an OWL ontology
- Product of W3C WS-Desc WG
- Jacek the main editor



# Presentation Overview

- Intentions of DERI
- Mapping details
- Differences from component model
- Logistical problems
- Future work



# Intentions of DERI

- Use of WSDL information in semantic data
  - Grounding WSMO directly to components
- Visibility of DERI
  - Commitment to Semantic Web Services

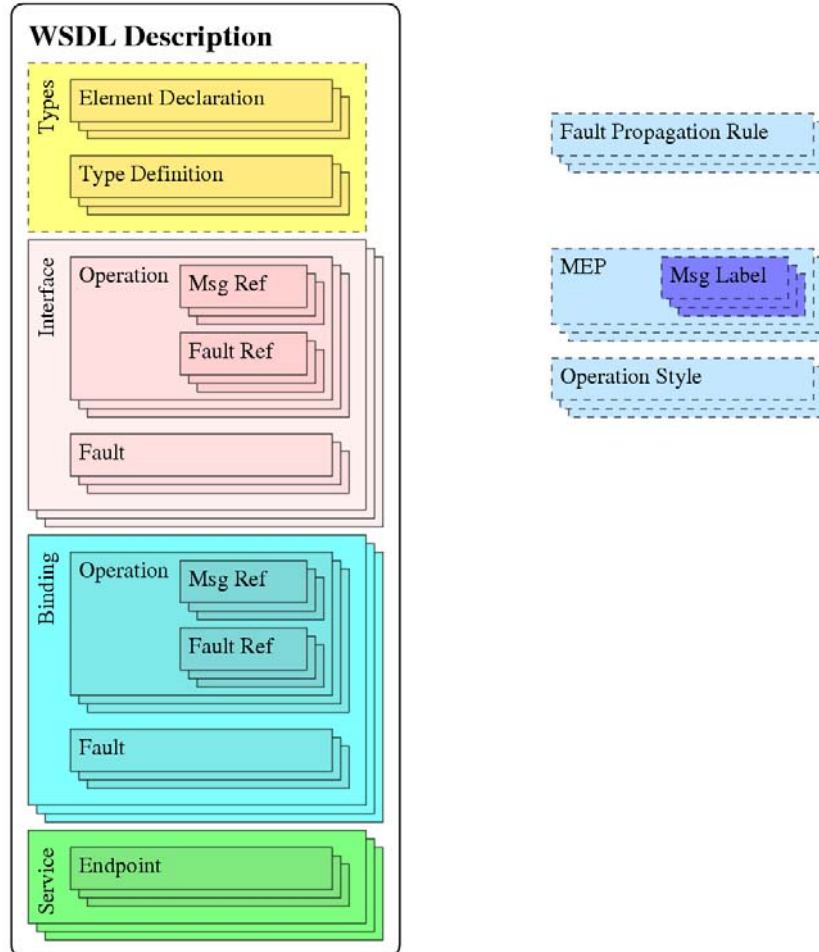


# WSDL RDF Mapping Structure

- Core components
- Extensibility
- Message Exchange Patterns
- Predefined extensions, operation styles
- SOAP and HTTP bindings

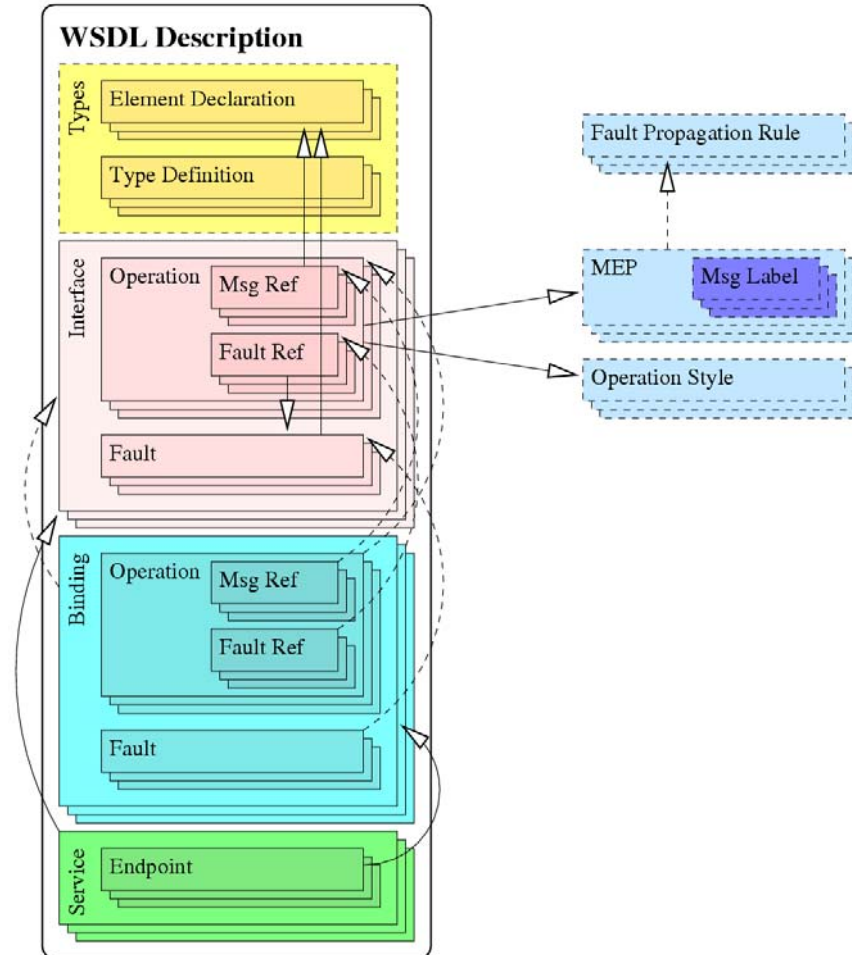


# WSDL Structure





# WSDL Structure With Links





# Core Components

- Description, Interface, Binding, Service
  - OWL classes
  - Generally referenced using properties with the same name, but lowercase
  - Unsure about namespace(s)
- Interface contains operations, faults
  - Operation contains message references
  - Operation has MEP, styles
  - Ops and faults point to elements
- Binding follows Interface structure
  - Binding types (SOAP, WSDL) are rdf:type
- Service has interface, endpoints
  - Endpoint has binding, address





# WSDL Extensions

- Extension points
  - Like MEPs, operation styles, bindings
  - Extensions should provide mapping similar to ours
- Features and Properties
  - Use URIs and QNames, easy to model in RDF
- Generic extensions
  - Mandatory extensions must specify RDF mapping for their parent components
  - Known optional extension must specify RDF mapping for self
  - Unknown optional extension elements mapped to XML literal
  - Unknown extension attributes modeled as ExtensionAttribute class
- Documentation as XML literal



# Message Exchange Patterns

- Modeled more formally than in component model
  - Class MessageExchangePattern
  - An MEP *defines* a number of MessageLabels
- InterfaceOperation points to MEP
  - InterfaceMessageReference points to MessageLabel



# Operation Extensions

- Operation styles
  - Just URIs, trivial mapping to RDF
  - RPC style adds extension property
- Operation Safety (predefined extension)
  - SafeInteraction class for InterfaceOperations

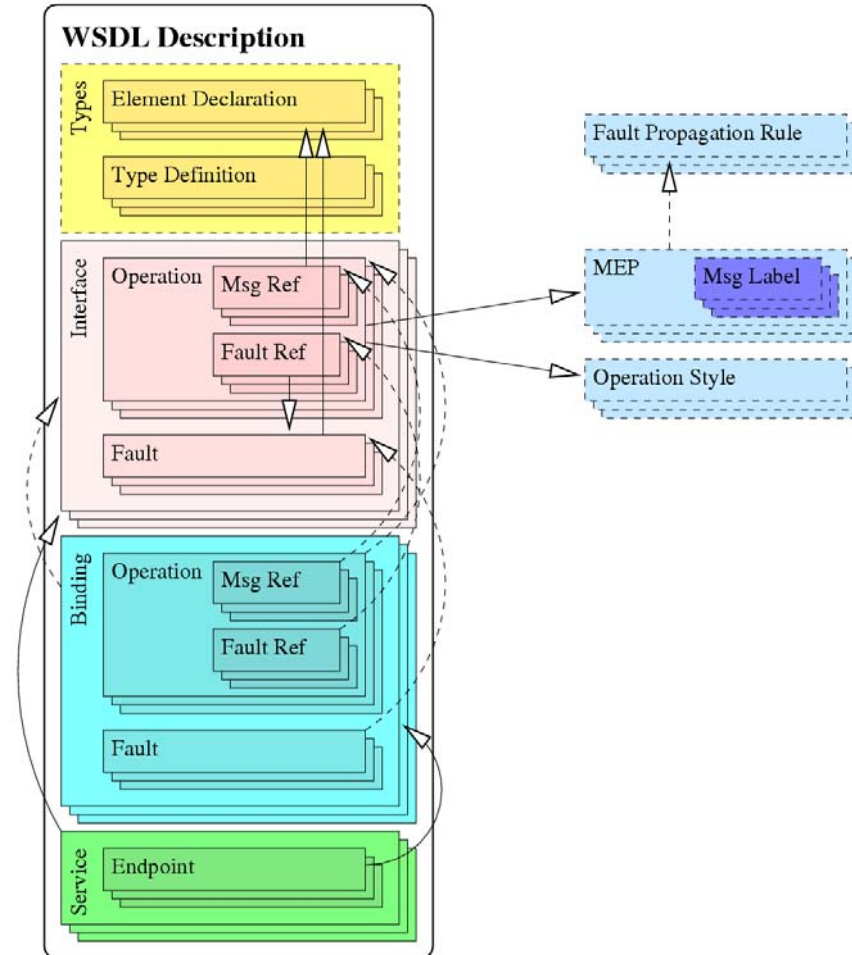


# SOAP and HTTP Bindings

- 2 classes and a bunch of properties
- SOAP binding class
  - Needs SOAPMessageExchangePattern class
    - ideally should be done by XMLP WG
- HTTP binding class
  - Plus subclass HTTPBindingWithCookies

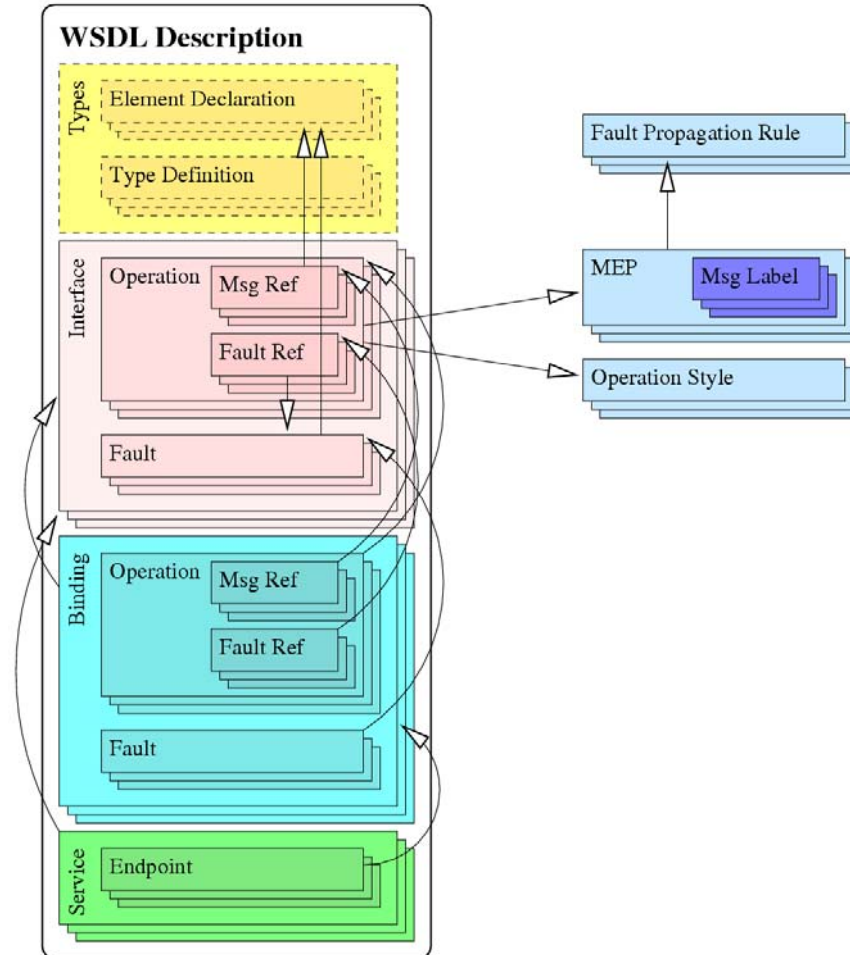


# WSDL Structure With Links (repeated)





# WSDL/RDF Structure





# Differences from Component Model

- RDF mapping works on **meaning** of the components
  - Mandatory extensions may change it, that's why such extensions must say how components are mapped
- WSDL has one Description, and RDF graph can have many
  - WSDL ontology is more lax than the component model (validation vs. inference)
- Component naming
  - WSDL has QName, RDF has URIs
  - QName deconstruction from URI is possible, not encouraged
  - Direct references between the components
- Imports and Includes are gone
- Enumerated properties into classes
  - Like message direction



# Logistical Problems

- Deliverable mandated by charter
  - Charter running out in Jan 2006 (extension likely)
- Only two group members work on it
  - Bijan Parsia main editor until this summer
  - Little progress, Jacek took over
  - First public draft 2005/11/4
- Unsure how to reach WG consensus
  - WG members cannot really review this
  - Few are interested
  - Unclear about the customer base





# Further Work

- Completing the mappings
- Maintenance wrt. issue resolutions
- CR (Candidate Rec) testing criteria?
  - CR = W3C beta testing
  - Two implementations
  - Turning WSDL/XML to WSDL/RDF enough?
- Follow-up in WSDL-S WG